

COMPUTER GAMES APPARATUS

US

The present invention relates to the field of computer games and, more particularly, to computer games played in a three dimensional (3D) computer graphics environment by multiple users.

Recently, multiple player computer games have become popular. In these games, each player controls one or more objects (e.g. people, cars etc.) in a common 3D environment. For example, in one implementation of a multiple-player game, each player is equipped with a terminal device, such as a personal computer (PC), with Internet access, to communicate with a server implementing a game via the Internet. The server stores three-dimensional object data and each player sends signals to the server via his PC to control his object(s) within the 3D environment defined by the object data. The server processes the object data, including making changes thereto in accordance with the signals received from all of the players, and generates image data showing the current status of the 3D environment, which is sent

to the players for displaying on their PC screens. The image data is generated by rendering the 3D object data from a viewing position defined for each player and which may be defined by that player. Image data for the
5 viewing position defined for a player is then sent to that particular player.

In view of the popularity of these games, it has become of interest to some people to observe the playing of a
10 game by a group of other players.

Figure 1 shows an arrangement which allows a plurality of users to play a game while under observation by an observer. This arrangement does not form part of the
15 present invention.

A server 10 is connected to the Internet 12 by an Internet connection 13. The server 10 is a computer on which is stored a computer program for implementation of
20 a computer game suitable for use by a plurality of players. That is, server 10 stores object data defining a three-dimensional game environment containing objects

to be controlled by players of the game and instructions defining how the object data is to be changed in response to signals from the players (which can be thought of as the game "rules"). A plurality of user terminals 14, 5 such as PCs, are connected by Internet connections 15 to the Internet 12.

Each of the Internet connections 13, 15 is operable to allow the bi-directional transmission of data between the 10 respective terminal 14 and the server 10. In that way, data can be passed from the terminal 14 to the server 10 to transmit player inputs to the game program, and data can be passed from the server 10 to the respective terminal 14 to update images displayed on the screen of 15 a user terminal 14.

An observer terminal 16 is illustrated, also connected to the server 10 via connection 18 to the Internet 12. However, in this case, the operator of the observer 20 terminal 16 takes no active part in the playing of the game embodied on the server 10 (that is, the observer cannot control any object in the stored object data).

Instead, the operator of the observer terminal 16 sends control commands to the server 10 to define a viewing position. In response to those signals, the server 10 renders the 3D object data from the defined viewing
5 position and sends signals back to the observer terminal 16 defining images to be displayed.

When an observer is added to the game environment, the server must generate additional image data by rendering
10 the object data from the viewing position defined for the observer. Such extra processing can cause players of the game to experience delays between registering inputs via their terminals and receiving confirmation of those inputs through images showing the changes being displayed
15 at their terminals. This situation produces "latency" which affects the ability of a player to retain a real-time link with the game environment.


If latency at a player's terminal reaches unacceptable
20 levels, the player will no longer be able to participate in the game effectively, since an image displayed on his screen at a particular point in real-time will relate to

a game position a considerable period of time earlier than the time in question.

Furthermore, the addition of more observers to the
5 network will further increase latency within the network, and removal of observers will reduce latency. Such increases and reductions in latency can be as distracting to actual players as the actual magnitude of the latency.

10 To address the problem of latency, the number of players and/or observers of a game is usually limited and/or the computer power of the server is increased. However, this approach can be inconvenient and expensive.


15 The present invention provides a computer graphics apparatus or method, in which data is stored defining a three-dimensional game environment and the data is changed in response to signals from users defining changes to objects in the environment, wherein data
20 relating to at least one view of the game environment is generated and output for broadcast to potential observers.



By broadcasting a view of the game environment, the game can be received by any number of observers and the number of observers does not affect the speed of operation of the processor on which the game environment is defined.

5

It will be appreciated that the term "broadcasting" is to be interpreted not only in the sense of television and radio broadcasting, where a signal is generated and broadcast into the air, but also in the sense of
10 transmission of signals along cables etc. from one transmitter to many potential recipients. Most broadcasting techniques do not require a broadcast signal to be addressed to particular recipients, and so any person in possession of appropriate equipment can obtain
15 access to information carried on a broadcast signal by tuning to the frequency (or channel) of the signal. In the case of TV broadcasting, the broadcast signal is largely unaffected by the number of receivers of the broadcast signal; in the case of broadcasting across the
20 Internet, provision is made by a broadcaster for a predetermined, but substantial, number of potential recipients, and that number is rarely reached except in



the event of broadcast information being of particular public interest.

The rendering means is preferably operable to generate
5 image data relating to a predefined number of different views of the game environment. In that way, a plurality of images, each from a different viewing position and/or direction, of the game environment can be broadcast to potential recipients, and the actual image to be viewed
10 by a recipient can be selected by that recipient from those broadcast.

Embodiments of the present invention will now be described, by way of example only, with reference to
15 Figures 2 to 8 of the accompanying drawings. In the drawings:

Figure 1 is a schematic diagram of an example of a known computer system for the implementation of a computer game
20 for a plurality of players and with the facility for the observation of the progress of the computer game by another party;

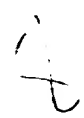


Figure 2 is a schematic diagram showing a computer network in a first embodiment of the present invention;

Figure 3 is a schematic diagram of the network server of
5 the computer network illustrated in Figure 2;

Figure 4 is a schematic block diagram of operational modules of the network server illustrated in Figure 3;

10 Figure 5 is a flow diagram showing functions of the 3D rendering pipeline illustrated in Figures 3 and 4;

Figure 6 is a schematic diagram of a user interface in the computer network illustrated in Figure 2;

15

Figure 7 is a schematic block diagram of a broadcast interface of the computer network illustrated in Figure 2;

20 Figure 8 is a schematic block diagram of a data encoder of the broadcast interface illustrated in Figure 7;

9

Figure 9 is a schematic block diagram of a receiver for use with the computer network illustrated in Figure 2; and

5 Figure 10 is a schematic diagram of an alternative embodiment of the present invention.

In a first embodiment, a computer network, as illustrated in Figure 2, comprises a network server 20, which is
10 connected to the Internet 22 by an Internet connection 21. The server 20 is under the control of software designed to implement a multiple player game, and has stored therein object data defining a 3D game environment and instructions on how the object data is to be changed
15 in response to signals from players. Server 20 receives signals from players of the game sent via the Internet 22, modifies the stored object data in response thereto, and transmits data defining images and sound back to the players via the Internet to show the current status of
20 the object data (game). These operations will be described in more detail below.

A plurality of user terminals 24 are connected to the Internet 22 by Internet connections 23. Each of the user terminals 24 comprises a personal computer 24 having access to the Internet. Alternatively, one or more of
5 the user terminals 24 can be implemented in other forms, such as a television combined with a set-top box for processing and Internet access. Each of the user terminals 24 is under the control of software which allows the user of the terminal to send instructions to
10 the server 20 defining how to change one or more objects in the game environment and to receive data defining images and sound from the server 20.

The computer network further comprises a broadcast
15 interface 26, which receives video and audio signals from the server 20, processes them, and broadcasts data defining video images on a number of different broadcast channels 38. The server 20 and broadcast interface 26 can be located remotely of each other.

20

More particularly, the server 20 has an output port 28, and the broadcast interface 26 has a data input port 30,

//

between which a data channel 32 is defined. In this embodiment, the data channel 32 comprises a hard-wired parallel connection for transfer of data along the channel in real time. However, other forms of data channel are possible. For example, the data channel may be defined by an output device, such as a disk drive, connected to the server 20 and an input device, such as another disk drive, connected to the broadcast interface 26, by which data can be stored on a storage medium, such as a disk, at the disk drive of the server 20, and then transferred to the broadcast interface 26 through the disk drive thereof (transfer of data in this manner being in non-real time). The data passed along the data channel 32 defines a series of video signals defining images of the 3D object data from a number of different viewpoints and audio signals corresponding to each series of images. In this embodiment, the viewpoints for the images sent from server 20 to broadcast interface 26 can be defined in the software instructions implementing the game stored on server 20, or can be specified by a system administrator operating the server 20.

The broadcast interface 26 has a control input port 34, to which is connected an input device 36. The input device 36 may comprise a keyboard and/or a position sensitive input device such as a mouse, tracker ball, or
5 a digitiser tablet and stylus etc. In combination therewith or in the alternative, the input device 36 may also comprise a video production desk or the like. By use of the user input device 36, a producer can create a broadcastable signal comprising a sequence of edited
10 portions of the image signals received from server 20.

The broadcast interface 26 broadcasts data on a plurality of channels 38, for reception by one or more receivers 40. In this embodiment, broadcast interface 26 transmits
15 image and sound data on a plurality of digital television channels through the atmosphere. However, one or more of the channels could comprise cabling. One or more of the channels could comprise at least one relay station which is adapted to receive signals, to refine those signals
20 such as by amplification and filtering, and to retransmit refined signals onward. Each channel transmits data for a respective different view of the game.

13

Each receiver 40 can tune to one of the broadcast channels and hence select the viewpoint from which the game is to be observed.

5 The server 20 will now be described further.

As illustrated in Figure 3, the server 20 comprises a central processing unit (CPU) 42, to which is coupled a memory 44. The memory 44 has coupled therewith a mass
10 storage device 56 for storage of data of which instant access is not required, and a disk drive 58 for receiving a storage medium, such as a disk 60, from which data can be transferred to the memory 44 and to which data can be transferred from the memory 44. The software, previously
15 described as implementing the multiple-player game, can be input into the server 20 via disk drive 58 by means of a storage medium, such as a disk, or by downloading from a database, for example via the Internet. The software is stored in memory 14 and routines thereof can be
20 accessed by the CPU 42. The memory 44 defines the three-dimensional object data such that the positions of all objects in the game environment are tracked.

14

Also coupled with the CPU 42 is an input/output port 46. A modem 48 is connected to the input/output port 46. The modem 48 is configured to connect to a telephone network (not shown) and the mass storage device 44 further has
5 software appropriate to the establishment of a connection to the Internet via the modem 48 and the telephone network.

The modem 48 allows the CPU 42 to receive signals from
10 players at terminals 24. The CPU 42 acts on those control signals in accordance with the software defining the game to make changes to the game environment in memory 44, and sends data defining images to be displayed at user terminals 24, via modem 48 and the Internet 12 to
15 the appropriate terminals 24. In this embodiment, server 20 sends data defining each pixel of an image. However, other types of data may be sent. For example, data defining changes to an image may be sent.

20 A frame buffer 50 is also provided, connected to the CPU 42. The frame buffer 50 is used to store frames of video images generated by the CPU for output to a visual

15

display unit (VDU) 52 connected therewith, so as to display images to an operator (system administrator) of server 20. The video images are preferably representative of views of the game as are displayed at the user terminals, and may also include a region allowing system commands and responses to be displayed. An input device 36 is connected via a control input port 34 of the CPU 42, to enable a system administrator to input system commands to the network server 20. As noted above, those commands and any responses thereto can be displayed on the VDU 52.

The CPU 42 further has connected therewith an output port 28, as previously noted with reference to Figure 2, for outputting data to broadcast interface 26.

It will be appreciated that, since the server 20 includes components standard to a personal computer, the server can comprise a personal computer. Moreover, the server 20 can also comprise one of the user terminals 24.

With reference to Figure 4 of the drawings, in use,

1/1

server 20 is programmed to provide functional units which include a game processor 110 and a rendering pipeline 112.

5 The game processor 110 is adapted to receive control signals from, and send data signals to, user terminals 24 in the playing of a multi-player computer game. The game processor 110 stores the object data to describe the three-dimensional game environment, including objects
10 whose positions can be controlled by the players of the game using their user terminals 24. The object data, representing playing positions and the game environment, can be changed by the game processor 110 acting in accordance with control signals from the various players
15 operating user terminals 24. In return, the game processor 110 responds by sending signals to each user terminal 24 defining a video image to be observed by the player at that user terminal 24. This allows the player at a user terminal 24 to interact with the game processor
20 110.

The rendering pipeline 112 is of standard form, for

5

example as described in "Computer Graphics Principles and Practice" by Foley, Van Dam, Feiner and Hughes, 2nd edition Addison-Wesley Publishing Company, ISBN 0-201-12110-7 pages 806 to 812.

5

The game server 110 and the 3D rendering pipeline 112 are connected so that object data defining the playing environment can be read by the 3D rendering pipeline 112. The rendering pipeline 112 converts the object data
10 relating to the three-dimensional playing environment into image data representing a plurality of views defined at viewpoints within the playing environment. Views are defined at viewpoints selected by each of the players (which may be limited by the game rules to be the
15 position of the player's character in the 3D game environment so that each player effectively sees the same thing as his character), and further views are defined at a plurality of other locations in the game environment. These further views can be predefined by the software
20 controlling the game, or they can be determined through the input of system commands by the system administrator via input device 36. These further views may be fixed or

12

may vary.

Image data defining the view selected by each player is returned from the rendering pipeline 112 to game processor 110 for transmission to the user terminals 24.

Image data defining the further views and also all of the views defined by the players is delivered to the output port 28 of the server 20, and from there via the data channel 32 to the input port 30 of the broadcast interface 26. In this embodiment, data defining each pixel of each image is output from server 20 to broadcast interface 26. However, other forms of data may be transferred. For example, data defining the changes to a previous image in the sequence may be sent to reduce the amount of data being transformed.

It will be understood that the delivery of image data to the broadcast interface 26 need not take place in real time. For example, if "live" broadcasting of video signals relating to the image data is not required, the image data can be sent to the disk drive 58 and written

19

to a disk 60, which can then be transferred and loaded into the broadcast interface 26.

Figure 5 illustrates a series of steps which describe the operation of the 3D rendering pipeline 112.

It will be noted that the series of steps comprises an endless loop. This is because the steps are carried out as long as the game is in operation, and is not affected by interrupts from external sources. In a first step S1, the 3D rendering pipeline 112 refers to the object data defined by the game server 110 and the viewpoints from which views are to be rendered. Once that information has been acquired, in step S2 the various views of the 3D environment defined by the object data are rendered by the 3D rendering pipeline 112 from the required viewpoints.

Then, in step S3, the data defining the images for display at user terminals is sent back to the game processor 110 from where they are sent via the Internet connections to the user terminals 24.

22

Finally, in step S4, the data defining all of the images produced in the rendering pipeline 112 is output to the output port 28 of the network server 20, from where it is sent to the input port 30 of the broadcast interface 26.

5

Once these steps have been completed, the procedure returns to the first step S1, and is repeated. In this embodiment, the procedure is repeated every 25 milliseconds, thereby producing 40 sets of new views every second, but other frame rates are, of course, possible.

10

The components of a user terminal 24 will now be described.

15

As illustrated in Figure 6, a user terminal comprises a personal computer 24, comprising a central processing unit (CPU) 60 coupled with a memory 62. The memory 62 has coupled therewith a mass storage device 76 and a disk drive 78. The mass storage device enables the storage of data therein which need not be available for instant access by the CPU 60, and the disk drive 78 is adapted to

receive a disk 80 for exchanging data between the disk 80 and the memory 62. The mass storage device 76 includes software implementing a user interface on the terminal 24 for interaction between the player and the user terminal

5 24.

Also coupled with the CPU 60 is an input/output port 64, coupled with a modem 66 to allow access to the Internet via a telephone line (not shown). The CPU 60 has an

10 input device 68 connected thereto, the input device 68 comprising a keyboard, and/or a position tracking device as described previously. The CPU 60 is connected to a frame buffer 70, which stores at least one frame of a video signal therein for output to a visual display unit

15 (VDU) 72 where an image can be displayed, or to a video tape recorder 74 where a video signal can be recorded for playback later. A sound driver 71 is connected to the CPU 60, to receive signals containing data defining audio signals. The sound driver 71 is connected to the video

20 recorder 74 for recording of the audio signals, and to a loudspeaker 73 associated with the visual display unit 72, for playback of audio signals in synchronisation with

video signals displayed on the visual display unit 72.

The user terminal 24 receives image data and associated audio signals via the Internet from the game processor 110 of server 20, and converts that data into image frames and audio signals. The frames are sent to the frame buffer 70 and the audio signals are sent to the sound drive 71. Moreover, the player using a user terminal 24 can observe the image displayed on the VDU 72, and make inputs via the keyboard 68. Those inputs can be representative of commands for controlling the game, and so the inputs are converted by the CPU 60 into control signals which are sent along the Internet to the game processor 110.

15

The components of the broadcast interface 26 will now be described.

As illustrated in Figure 7, the broadcast interface 26 includes a producer selector 114 and a user interface 116.

The producer selector 114 and user interface 116 are arranged to enable an operator (producer) to select images to be broadcast from the available views (and therefore perform the functions of a conventional TV image data selector as operated by a producer when images from a number of cameras are available).

More particularly, producer selector 114 receives all of the image data from the input port 30 for all views generated by the server 20, and outputs image data for a single stream of images selected from the available views. The producer selector 114 is connected to the user interface 116, and control signals generated in the user interface are operative to determine the selection of the images to be output from the producer selector 114.

The user interface 116 has connected thereto an input device 32, such as a video production desk, of conventional type. The original images received from the input port 30 and the images output from the producer selector 114 are fed to the user interface 116. The user

interface 116 processes all of the image data to display the images on an array of visual display units 118, each display unit displaying the images for one viewing position and one of the display units displaying the
5 images selected by the operator.

In that way, the producer selector 114 can be operated to select different views of the 3D game environment and, with sufficient skill of the operator, a meaningful
10 sequence of "camera angles" (views) of the 3D playing environment can be generated, thereby emulating a sequence of camera angles as might be provided in a real television production.

INS
C1
15 ~~It will be appreciated that the array of visual display~~
~~units 118 may be replaced by a simple visual display unit~~
~~displaying an image comprising a plurality of regions,~~
~~each region containing an image otherwise displayed on~~
~~one of the array of visual display units 118 illustrated~~
20 ~~in Figure 7.~~

Also connected to the user interface 116 is a microphone 117, through which a commentary can be added to the video

25

signals. For instance, if the game is based on a sport or similar activity, a television personality connected with that sport could be engaged to provide comment on the progress of the game. In that way, commentary may be
5 provided in a local language to a game actually played in a foreign country. The commentary may be provided as an overlay over existing audio signals attached to the image data, or in replacement thereof.

10 The images received from the input port 30 and the images output from the producer selector 114 are fed to a MPEG2 encoder 120, which generates compressed video signals.

Each of the compressed video signals is then broadcast by
15 satellite communication and/or cable and/or electromagnetic radiation transmission on a separate digital television channel, so that they can be received by any member of the public anywhere in the world.

20 In this embodiment, for illustration purposes, the rendering pipeline 112 produces image data for seven different viewing positions, and the image data for each

26

of these positions is broadcast on a separate channel. In addition, image data for images selected by the producer from the seven different viewing positions is produced by producer selector 114 and broadcast on an eighth channel. Accordingly, in this embodiment, MPEG2 encoder 120 is an eight channel encoder.

The structure of the channels of data output by the MPEG2 encoder 120 can be more clearly observed in Figure 8. That figure shows the input of image data to the encoder 120, and the output of eight channels, referenced as channels 1 to 7 and channel p. Channel p comprises the producer selected channel, being a stream in real time selected from portions of the signals on the other seven channels.

Figure 9 shows a receiver 40 for receiving the channels output by the eight channel MPEG2 encoder. In practice, the receiver is operable to receive digital television signals in general. The receiver includes a channel selector 122, which receives the eight channels p and 1 to 7. The channel selector receives signals from an

2

input device 124 such as a remote control unit, and selects one of the eight signals for viewing. The selected channel is output to an MPEG2 decoder 126, which produces a signal adapted to be received by a television receiver 128. The channel selector 122 and the MPEG2 decoder 126 can be incorporated into a set top box for domestic purposes.

The output of the MPEG2 decoder 126 is connected to a television 128 and to a video tape recorder (VTR) 130. The VTR 130 can be used to record the output, which is a video/audio signal, on a video tape 132. Further recordings may then be made by making recordings directly or indirectly from tape 132.

15

In accordance with the foregoing, a multi-player game can be played in a three-dimensional playing environment, and the state of the game can be viewed by any number of observers from one of a selection of viewpoints, without influencing the speed at which the game can be played. Moreover, a commentary may be added, for reception by an observer.

24

Figure 10 shows a computer network in a second embodiment. The network comprises five terminals 150. Each terminal 150 consists of a personal computer, and each terminal 150 is connected to a communication channel 152, such as a LAN etc. A broadcast interface 154 is also connected to the communication channel 152, and monitors communication between the terminals 150 on the channel 152.

10 In use, each terminal 150 is provided with its own copy of the software as stored on server 20 in the first embodiment, that is object data defining the three-dimensional game environment and processor implementable instructions defining how the object data is to be
15 modified in response to player instructions. Inputs made by a user of a terminal 150 cause changes to be made to the object data stored on that terminal (rather than the object data stored on a server, as in the first embodiment). Those changes are communicated to all of
20 the terminals 150 in the network, by means of the communication channel 152, and the object data stored locally on those terminals are changed accordingly. The

broadcast interface 154 also has a copy of the object data, which it updates in accordance with the signals received from all of the terminals 150.

5 The broadcast interface 154 includes a rendering pipeline that renders object data into image data from predetermined viewpoints. The image data is then broadcast to potential recipients along a broadcast channel 156.

10

In the second embodiment, it should be noted that the addition of further terminals to the network may reduce the efficiency of the communication channel 154. However, since data is broadcast to observers, the actual
15 number of observers is not relevant to the effectiveness of the communication between the terminals 150 via the communication channel 152.

A third embodiment of the invention will now be
20 described. The third embodiment is the same as the first embodiment, but with the following changes.

30

In the third embodiment of the invention, the network server 20 has stored therein object data defining a 3D game environment and instructions on how the object data is to be changed in response to signals from players, as in the first embodiment. However, in addition, each of the user terminals 24 has stored therein a copy of the object data defining the 3D game environment. Server 20 receives signals from players of the game at terminals 24 sent via the Internet 22, modifies the object data stored in the server in response thereto, and transmits data defining changes to the stored object data back to the user terminals 24 (rather than image data as in the first embodiment). Upon receipt of the change data from the server 20, each user terminal 24 updates the copy of the object data stored at the terminal accordingly. Each user terminal then renders an image from the object data stored at the terminal from a selected viewpoint within the 3D game environment. For each player, the position and direction of the view within the 3D game environment is transmitted back to the server 20, which then constructs data defining images at those viewpoints (and other viewpoints) so that data can be broadcast by the

31

broadcast interface 26. Of course, if the views of the
players are not to be broadcast, then it is unnecessary
for each terminal 24 to send information to the server
defining the viewing parameters of the player at the
5 terminal.

SECRET

32